



Second Semester Examination
2016/2017 Academic Session

June 2017

CPT212 – Design & Analysis of Algorithms
[Reka Bentuk & Analisis Algoritma]

Duration : 2 hours
[Masa : 2 jam]

INSTRUCTIONS TO CANDIDATE:

[ARAHAN KEPADA CALON:]

- Please ensure that this examination paper contains **THREE** questions in **EIGHT** printed pages before you begin the examination.

*[Sila pastikan bahawa kertas peperiksaan ini mengandungi **TIGA** soalan di dalam **LAPAN** muka surat yang bercetak sebelum anda memulakan peperiksaan ini.]*

- Answer **ALL** questions.

*[Jawab **SEMUA** soalan.]*

- You may answer the questions either in English or in Bahasa Malaysia.

[Anda dibenarkan menjawab soalan sama ada dalam bahasa Inggeris atau bahasa Malaysia.]

- In the event of any discrepancies, the English version shall be used.

[Sekiranya terdapat sebarang percanggahan pada soalan peperiksaan, versi bahasa Inggeris hendaklah diguna pakai.]

1. The multiplications of 2 numbers in base 10 can be done using the **standard multiplication algorithm**. The algorithm is explained below:

Step 1: Multiply each digit in the multiplicand by each digit of the multiplier, keep the partial product and carriers if there is any (e.g. $3 \times 8 = 24$, "4" is the partial product and "2" is the carrier).

Step 2: Repeat Step 1 for all the digits in the multiplier.

Step 3: Add up all the properly shifted partial product and carriers.

Pendaraban 2 nombor dalam asas 10 boleh dilakukan dengan menggunakan algoritma pendaraban piawai. Algoritma tersebut dijelaskan seperti di bawah:

Langkah 1: Darabkan setiap angka dalam multiplicand dengan setiap angka pengganda, simpan produk separa dan pembawa jika ada (cth $3 \times 8 = 24$, "4" ialah produk separa dan "2" ialah pembawa).

Langkah 2: Ulangi Langkah 1 bagi semua digit dalam pengganda.

Langkah 3: Tambah semua produk separa dan pembawa yang telah dialihkan.

Figure below shows an example where the multiplicand '52301' is multiplied by the multiplier '380' using the standard multiplication algorithm.

Rajah di bawah menunjukkan satu contoh di mana multiplicand '52301' didarab dengan pengganda '380' dengan menggunakan algoritma pendaraban piawai.

	52301	
x	380	
	00000	partial products for (52301 x 0) / produk separa bagi (52301 x 0)
	00000	carriers for (52301 x 0) / pembawa bagi (52301 x 0)
	06408	partial products for (52301 x 8) / produk separa bagi (52301 x 8)
	41200	carriers for (52301 x 8) / pembawa bagi (52301 x 8)
	56903	partial products for (52301 x 3) / produk separa bagi (52301 x 3)
+	10000	carriers for/pembawa bagi (52301 x 3)
	19874380	

- (a) Write a function **`void getAllProductAndCarrier(int[] multiplicand, int[] multiplier)`** in Java to implement Step 1 and Step 2 of the standard multiplication algorithm. Assume **`int Math.getPartialProduct(int multiplicand, int multiplier)`** gives you the partial product value and **`int Math.getCarrier(int multiplicand, int multiplier)`** gives you the carrier value. E.g. `Math.getPartialProduct(3, 8)` returns 4, and `Math.getCarrier(3, 8)` returns 2.

*Tulis satu fungsi **`void getAllProductAndCarrier(int[] multiplicand, int[] multiplier)`** dalam Java untuk melaksanakan Langkah 1 dan Langkah 2 bagi algoritma pendaraban piawai. Andaikan **`int Math.getPartialProduct (int multiplicand, int multiplier)`** memberikan anda nilai produk separa dan **`int Math.getCarrier(int multiplicand, int multiplier)`** memberikan anda nilai pembawa. Contohnya, `Math.getPartialProduct(3,8)` memberi 4, dan `Math.getCarrier(3, 8)` memberi 2.*

(36/100)

- (b) Name **two (2)** primitive operations used in the standard multiplication algorithm?

*Namakan **dua (2)** operasi primitif yang digunakan dalam algoritma pendaraban piawai?*

(4/100)

- (c) If the length of multiplicand is n , and the length of the multiplier is m , what is the asymptotic time complexity in big O for the standard multiplication algorithm?

Jika panjang multiplicand ialah n , dan panjang pengganda ialah m , apakah kekompleksan asimptot dalam O besar bagi algoritma pendaraban piawai.

(10/100)

- (d) If you implement the **`int Math.getPartialProduct(int x, int y)`**, describe the way you would implement the method, so that

*Jika anda melaksanakan **`int Math.getPartialProduct(int x, int y)`**, terangkan cara anda akan melaksanakan fungsi tersebut, supaya*

- (i) it has good time complexity.

ia mempunyai kekompleksan masa yang baik.

- (ii) it has good space complexity.

ia mempunyai kekompleksan ruang yang baik.

(20/100)

- (e) In the computational complexity theory, which time complexity class the multiplication problem belongs to? What is the definition of that class?

Dalam teori kerumitan pengiraan, masalah pendaraban tergolong dalam kelas kerumitan apa? Apakah definisi kelas tersebut?

(10/100)

- (f) Explain with a figure what is a Turing machine and why in the analysis of complexity, Turing machine is used?

Terangkan dengan gambar rajah apa itu mesin Turing dan mengapa dalam analisis kekompleksan, mesin Turing digunakan?

(20/100)

2. (a) Show the following three types of binary trees for the following list of keys:

Tunjukkan tiga jenis pepohon perdua berikut bagi senarai kunci berikut:

ALGORITHMS.

- (i) Binary Search Tree.

Pepohon Gelitaran Perdua.

- (ii) AVL Tree.

Pepohon AVL.

- (iii) Heap (max).

Timbunan (max).

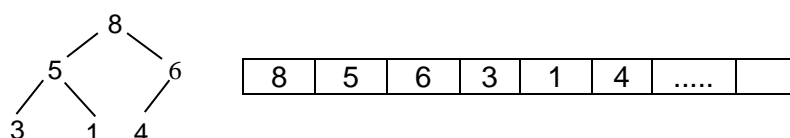
For n keys, what is the height of the best case for each tree?

Bagi n kunci, apakah ketinggian kes terbaik untuk setiap pepohon?

(15/100)

- (b) Write a pseudocode that will output the sum of all elements of a heap. For example for the following heap and its array representation the output will be 27.

Tulis pseudokod yang akan memaparkan hasil tambah semua unsur sesebuah timbunan. Sebagai contoh timbunan berikut dan perwakilan tatasusunannya akan dioutputkan sebagai 27.



(15/100)

- (c) Insert the following keys in the given order into a (2-4) tree that is initially empty:
5 16 22 45 2 10 18 30 50 12 1

Sisipkan kunci-kunci berikut dalam tertib yang diberikan ke dalam pepohon (2-4) yang pada awalnya kosong: 5 16 22 45 2 10 18 30 50 12 1

(20/100)

- (d) (i) A digraph application frequently requires the process of determining the number of edges in a given graph. What is the most efficient representation for this application? Consider only adjacency matrix and adjacency lists representations and justify your answer.

Sebuah aplikasi graf berarah kerap memerlukan proses penentuan bilangan tepi yang terdapat di dalam sesebuah graf yang diberikan. Apakah perwakilan yang paling cekap bagi aplikasi ini? Pertimbangkan sama ada perwakilan matriks kesebelahan atau senarai kesebelahan sahaja dan jelaskan jawapan anda.

- (ii) Write a pseudocode that will calculate the number of edges in a given directed node G using the representation that you have chosen in Question 2(d)(i) above.

Tulis satu pseudokod untuk mengira bilangan tepi yang terdapat di dalam graf G dalam sebuah graf berarah yang diberikan dengan menggunakan perwakilan yang anda pilih dalam Soalan 2(d)(i) di atas.

(20/100)

- (e) There are eight small islands in a lake, and the government wants to build seven bridges to connect them so that each island can be reached from any other one via one or more bridges. The cost of constructing a bridge is proportional to its length. The distances between pairs of islands are given in the following table.

Terdapat lapan pulau kecil dalam sebuah tasik, dan kerajaan ingin membina tujuh jambatan untuk menghubungkannya supaya setiap pulau boleh dikunjungi dari mana-mana pulau melalui satu atau lebih jambatan. Kos pembinaan satu jambatan bergantung kepada panjang. Jarak di antara dua pulau diberikan dalam jadual di bawah.

Table 1: The Distance Between Pairs of Island
Jadual 1: Jarak Antara Dua Pulau

	1	2	3	4	5	6	7	8
1	-	240	210	340	280	200	345	120
2	-	-	265	175	215	180	185	155
3	-	-	-	260	115	350	435	195
4	-	-	-	-	160	330	295	230
5	-	-	-	-	-	360	400	170
6	-	-	-	-	-	-	175	205
7	-	-	-	-	-	-	-	305
8	-	-	-	-	-	-	-	-

- (i) Draw a graph to represent the table above.

Lukis satu graf untuk mewakili jadual di atas.

- (ii) Describe an algorithm/pseudocode to build the seven bridges.

Huraikan satu algoritma/pseudokod untuk membina tujuh jambatan tersebut.

- (iii) What is the total cost of the bridges you have constructed?

Berapakah jumlah kos pembinaan kesemua jambatan tersebut?

(30/100)

3. (a) Given a hash function, $\text{hash}(\text{key}) = \text{the middle letter of the key} \% \text{size of table}$, and the keys are assumed to be of the same length and the length of the key is an odd number.

Diberikan sebuah fungsi cincangan, $\text{hash}(\text{key}) = \text{huruf tengah key} \% \text{saiz jadual}$, dan kunci-kunci berkenaan dianggap mempunyai panjang yang sama dan panjang kunci adalah satu nombor ganjil.

- (i) Is the hash function a good hash function? Justify your answer. Give also a pattern that hashes into the same location for the hash function.

Adakah fungsi cincangan berkenaan baik? Jelaskan jawapan anda. Beri juga pola yang mencincang ke dalam lokasi yang sama bagi fungsi berkenaan.

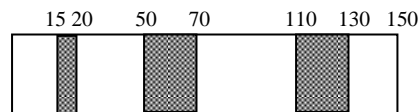
- (ii) Write a function in Java that implements the above hash function.

Tulis fungsi dalam Java yang melaksanakan fungsi cincangan di atas.

(20/100)

- (b) Given the following heap:

Diberi timbunan berikut:



- (i) Give a series of memory allocations using the best-fit algorithm in order to show one of the disadvantages of the algorithm namely the remaining parts after returning the required size are small and practically unusable.

Beri satu siri peruntukan ingatan menggunakan algoritma penyesuaian terbaik supaya satu daripada keburukan algoritma berkenaan ditunjukkan iaitu bahagian yang tinggal selepas memulangkan saiz yang diperlukan adalah kecil dan tidak boleh digunakan secara praktik.

- (ii) Give a series of memory allocations using the worst-fit algorithm in order to show one of the advantages of the algorithm namely the method prevents fragmentation by avoiding/delaying creation of small blocks.

Beri satu siri peruntukan ingatan menggunakan algoritma penyesuaian terburuk supaya satu daripada kebaikan algoritma berkenaan ditunjukkan iaitu menghalang penyerpihan dengan mengelak/melengahkan penciptaan blok-blok kecil.

(30/100)

- (c) A pseudocode for a simple string matching approach is given below:

Pseudokod bagi pendekatan pemadanan rentetan yang mudah diberi di bawah:

```
bruteForceStringMatching(pattern P, text T)
  i = 0;
  while i ≤ |T| - |P|
    j = 0;
    while Ti == Pj and j < |P|
      i++;
      j++;
    if j == |P|
      return match at i - |P|;
    i = i - j + 1;
  return no match;
```

- (i) What is the purpose of the inner **while** loop?

*Apakah tujuan gelung **while** dalam?*

- (ii) What is the purpose of the outer **while** loop?

*Apakah tujuan gelung **while** luar?*

- (iii) What is the worst case complexity of this algorithm? Show how you obtain the answer.

Apakah kekompleksan kes terburuk algoritma ini? Tunjuk bagaimana anda mendapatkan jawapannya.

- (iv) Give one example of pattern P and text T that will give the worst case situation. Show the tracing of the example that you have given.

Beri satu contoh pola P dan teks T yang akan memberikan situasi kes terburuk. Tunjuk surihan contoh yang anda berikan.

(35/100)

- (d) You are given five letters A, B, C, D, and E with probabilities 0.05, 0.25, 0.4, 0.2, and 0.1 respectively. Create a Huffman tree for the letters.

Anda diberi lima huruf A, B, C, D, dan E dengan kebarangkalian masing-masing 0.05, 0.25, 0.4, 0.2, dan 0.1. Cipta satu pepohon Huffman untuk huruf-huruf berkenaan.

(15/100)